Package 'Harshlight'

September 24, 2012

Version 1.28.0

Date 2011-02-09

Title A ''corrective make-up'' program for microarray chips

Author Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M. Wittkowski, Marcelo O. Magnasco

Maintainer Maurizio Pellegrino <mpellegri@rockefeller.edu>

Depends R (>= 2.10)

Imports affy, altcdfenvs, Biobase, stats, utils

Description The package is used to detect extended, diffuse and compact blemishes on microarray chips. Harshlight automatically marks the areas in a collection of chips (affybatch objects) and a corrected AffyBatch object is returned, in which the defected areas are substituted with NAs or the median of the values of the same probe in the other chips in the collection.
The new version handle the substitute value as whole matrix to solve the memory problem.

License GPL (>= 2)

URL http://asterion.rockefeller.edu/Harshlight/

biocViews Microarray, QualityControl, Preprocessing, AffymetrixChip,ReportWriting

R topics documented:

HarshComp *a blemish detection program for microarray chips: extended and compact defects only*

Description

Harshlight automatically detects and masks blemishes in microarray chips of class AffyBatch

Usage

```
HarshComp(affy.object, my.ErrorImage = NULL, extended.radius = 10,
compact.quant.bright = 0.025, compact.quant.dark = 0.025,
compact.size.limit = 15, compact.connect = 8, compact.pval = 0.01,
percent.contiguity = 50, report.name = 'R.report.ps',
na.sub = FALSE, interpolate = TRUE)
```

Arguments

affy.object	An AffyBatch object containing two or more chips.
my.ErrorImage	A batch of ErrorImages obtained through other programs. The error images must be in a matrix format, in which the first index represents each cell in the matrix and the second index represents the chip number. By default, the program calculates the error images for the batch of chips affy.object as described in Suarez-Farinas M et al., BMC Bioinformatics - 2005. If a batch of error images is provided, the affy.object is also required.
extended.radius	Radius of the median kernel used to identify extended defects on the chip.
compact.quant.bright, compact.quant.dark	Quantiles of the Error Image used to declare outliers. Values bigger than the '(1 - compact.quant.bright)' percentile are bright outliers, while values smaller than the 'compact.quant.dark' percentile are dark outliers. The two quantiles are used to detect compact defects. Set it to 0 to turn compact defect detection off.
compact.size.limit	Minimum size for clusters to be considered defects. If 0, all the clusters identified will be considered defects, if their size is significantly bigger than the one expected by chance (see also compact.pval).
compact.connect	Defines the neighbourhood of a pixel, used to connect outliers into clusters. If 4, the neighbourhood contains the pixels that are adjacent of a pixel of reference, on the vertical or horizontal axis. If 8, the neighbourhood contains all the 8 pixels sorrounding the pixel of reference. If a connectivity of 4 is used, clusters that are connected only through an edge will be considered as separate clusters. In this case, the single clusters could be eliminated because their size does not exceed compact.size.limit or diffuse.size.limit. Therefore, we suggest to use a connectivity of 8.
compact.pval	Threshold for compact defect size. This is the maximum probability accepted to find a cluster of the same size by chance. If 1, a cluster is considered a compact defect if it is bigger than the value of compact.size.limit.

`percent.contiguity`
: Minimum percentage of area density for defects to be considered compact. If 0, every compact defect found will be eliminated before searching for diffuse defects. Though possible, avoid using less than 20; otherwise diffuse defects might not be identified properly.

`report.name`
: Name of the PostScript file in which to save the final report. If report.name is set to "", no report will be written.

`na.sub`
: If TRUE, the intensity values of the input affyBatch that are affected by defects will be changed in NA. If FALSE, the values will be substituted with the median of the intensity values of the other chips.

`interpolate`
: This option is only used if the value of compact.quant.bright or compact.quant.dark is not among those tabulated (density of outliers = 0.01, 0.02, 0.05, 0.10, 0.20, 0.25, 0.30, 0.40; chip size = 534x534, 640x640, 712x712). If TRUE, the cluster size distribution under the null hypothesis of spatially randomly distributed outliers is derived from simulated values through interpolation. If FALSE, the distribution is simulated for the input value of density of outliers (compact.quant.bright/compact.quant.da and the specific chip size. The program runs 100.000 simulations by default.

Value

HarshComp is used to detect extended and compact defects only.

`AffyBatch object`
: The input AffyBatch object, whose intensity values corresponding to defected areas are substituted either by NA or by the median of the chip's values (depending on na.sub).

`Report`
: For each AffyBatch analyzed, a report is written as a PostScript file (see also report.name).

Author(s)

Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M. Wittkwosky, Marcelo O. Magnasco <mpellegri@rockefeller.ed

References

http://asterion.rockefeller.edu/Harshlight/

Harshlight: a "corrective make-up" program for microarray chips, Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M Wittkowski and Marcelo O Magnasco, BMC Bioinformatics 2005 Dec 10; 6(1):294

"Harshlighting" small blemishes on microarrays, Suarez-Farinas M, Haider A, Wittkowski KM., BMC Bioinformatics. 2005 Mar 22;6(1):65.

`HarshExt` *a blemish detection program for microarray chips: extended defects only*

Description

Harshlight automatically detects and masks blemishes in microarray chips of class `AffyBatch`

Usage

```
HarshExt(affy.object, my.ErrorImage = NULL, extended.radius = 10)
```

Arguments

affy.object	An AffyBatch object containing two or more chips.
my.ErrorImage	A batch of ErrorImages obtained through other programs. The error images must be in a matrix format, in which the first index represents each cell in the matrix and the second index represents the chip number. By default, the program calculates the error images for the batch of chips affy.object as described in Suarez-Farinas M et al., BMC Bioinformatics - 2005. If a batch of error images is provided, the affy.object is also required.
extended.radius	Radius of the median kernel used to identify extended defects on the chip.

Value

HarshExt is used to detect only extended defects on the surface of the chip. It does not detect compact or diffuse defects (see the help page for Harshlight).

Author(s)

Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M. Wittkwosky, Marcelo O. Magnasco <mpellegri@rockefeller.ed

References

http://asterion.rockefeller.edu/Harshlight/

Harshlight: a "corrective make-up" program for microarray chips, Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M Wittkowski and Marcelo O Magnasco, BMC Bioinformatics 2005 Dec 10; 6(1):294

"Harshlighting" small blemishes on microarrays, Suarez-Farinas M, Haider A, Wittkowski KM., BMC Bioinformatics. 2005 Mar 22;6(1):65.

Harshlight *a blemish detection program for microarray chips: extended, diffuse, and compact defects*

Description

Harshlight automatically detects and masks blemishes in microarray chips of class AffyBatch

Usage

```
Harshlight(affy.object, my.ErrorImage = NULL, extended.radius = 10,
compact.quant.bright = 0.025, compact.quant.dark = 0.025,
compact.size.limit = 15, compact.connect = 8, compact.pval = 0.01,
diffuse.bright = 40, diffuse.dark = 35, diffuse.pval = 0.001,
diffuse.connect = 8, diffuse.radius = 10,
diffuse.size.limit = (3*3.14*(diffuse.radius**2)),
percent.contiguity = 50, report.name = 'R.report.ps', na.sub = FALSE,
interpolate = TRUE, diffuse.close = TRUE)
```

Arguments

affy.object	An AffyBatch object containing two or more chips.
my.ErrorImage	A batch of ErrorImages obtained through other programs. The error images must be in a matrix format, in which the first index represents each cell in the matrix and the second index represents the chip number. By default, the program calculates the error images for the batch of chips affy.object as described in Suarez-Farinas M et al., BMC Bioinformatics - 2005. If a batch of error images is provided, the affy.object is also required.
extended.radius	Radius of the median kernel used to identify extended defects on the chip.
compact.quant.bright, compact.quant.dark	Quantiles of the Error Image used to declare outliers. Values bigger than the '(1 - compact.quant.bright)' percentile are bright outliers, while values smaller than the 'compact.quant.dark' percentile are dark outliers. The two quantiles are used to detect compact defects. Set it to 0 to turn compact defect detection off.
compact.size.limit, diffuse.size.limit	Minimum size for clusters to be considered defects. If 0, all the clusters identified will be considered defects, if their size is significantly bigger than the one expected by chance (see also compact.pval).
compact.connect, diffuse.connect	Defines the neighbourhood of a pixel, used to connect outliers into clusters. If 4, the neighbourhood contains the pixels that are adjacent of a pixel of reference, on the vertical or horizontal axis. If 8, the neighbourhood contains all the 8 pixels sorrounding the pixel of reference. If a connectivity of 4 is used, clusters that are connected only through an edge will be considered as separate clusters. In this case, the single clusters could be eliminated because their size does not exceed compact.size.limit or diffuse.size.limit. Therefore, we suggest to use a connectivity of 8.
compact.pval	Threshold for compact defect size. This is the maximum probability accepted to find a cluster of the same size by chance. If 1, a cluster is considered a compact defect if it is bigger than the value of compact.size.limit.
diffuse.bright, diffuse.dark	Percentage of increase (bright) or decrease (dark) of the intensity value of a pixel compared to the expected intensity. Used to declare outliers to detect diffuse defects. The option to detect diffuse defects is turned off if the value is set to 0.
diffuse.radius	Radius of the mask used to identify diffuse defects on the chip. Inside this mask the binomial test is performed.
diffuse.pval	Significance for the binomial test during diffuse defects' detection.
percent.contiguity	Minimum percentage of area density for defects to be considered compact. If 0, every compact defect found will be eliminated before searching for diffuse defects. Though possible, avoid using less than 20; otherwise diffuse defects might not be identified properly.
report.name	Name of the PostScript file in which to save the final report. If report.name is set to '', no report will be written.
na.sub	If TRUE, the intensity values of the input affyBatch that are affected by defects will be changed in NA. If FALSE, the values will be substituted with the median of the intensity values of the other chips.

`interpolate` This option is only used if the value of compact.quant.bright or compact.quant.dark is not among those tabulated (density of outliers = 0.01, 0.02, 0.05, 0.10, 0.20, 0.25, 0.30, 0.40; chip size = 534x534, 640x640, 712x712). If TRUE, the cluster size distribution under the null hypothesis of spatially randomly distributed outliers is derived from simulated values through interpolation. If FALSE, the distribution is simulated for the input value of density of outliers (compact.quant.bright/compact.quant.da and the specific chip size. The program runs 100.000 simulations by default.

`diffuse.close` If TRUE, the whole area in which the diffuse defects are included is considered as a defect. If FALSE, only the outliers inside the area are considered defects.

Value

`AffyBatch object`
The input AffyBatch object, whose intensity values corresponding to defected areas are substituted either by NA or by the median of the chip's values (depending on na.sub).

`Report` For each AffyBatch analyzed, a report is written as a PostScript file (see also report.name).

Author(s)

Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M. Wittkwosky, Marcelo O. Magnasco <mpellegri@rockefeller.ed

References

http://asterion.rockefeller.edu/Harshlight/

Harshlight: a "corrective make-up" program for microarray chips, Mayte Suarez-Farinas, Maurizio Pellegrino, Knut M Wittkowski and Marcelo O Magnasco, BMC Bioinformatics 2005 Dec 10; 6(1):294

"Harshlighting" small blemishes on microarrays, Suarez-Farinas M, Haider A, Wittkowski KM., BMC Bioinformatics. 2005 Mar 22;6(1):65.

Examples

```
## To run the example, download the affybatch object example.rda
## from the website http://asterion.rockefeller.edu/Harshlight/

## Not run:
source("example.rda") ## this creates the object my.affybatch in your working environment
library(Harshlight)
harsh <- Harshlight(affy.object = my.affybatch, report.name = 'example.ps') ## The file example.ps will app

## Calculate expression measures using MAS5
mas.example <- mas5(my.affybatch)
mas.harsh <- mas5(harsh)
plot(log2(exprs(mas.example)),log2(exprs(mas.harsh)))

## End(Not run)
```

`sim` *blemish simulations from 100.000 random chips*

Description

This data set contains the probability distribution of the cluster size under the assumption of spatially randomly distributed outliers. The distribution depends on the chip size, the density of outliers and the definition of connectivity (see help(Harshlight)). The sets sim_n4_/sim_n8_ contain the results from 100.000 simulations (sim.pval), while the sets sim_4_/sim_8_ contain the parameters that are used to interpolate the probability for values of density of outliers and chip sizes not simulated (a, b). The data set is used by the package Harshlight and is not intended for direct use by the user.

Format

The number of occurrences of at least one cluster of a certain size in 100.000 chips with randomly distributed outliers.

Index